NASA TECHNICAL NOTE

NASA TN D-2257

C.1

NASA TN D-2257

CODING AN ANALOG VARIABLE FOR CONSTANT PERCENTAGE ERROR

by Rodger A. Cliff

Goddard Space Flight Center
Greenbelt, Maryland

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MARCH 1964

CODING AN ANALOG VARIABLE FOR CONSTANT PERCENTAGE ERROR

By Rodger A. Cliff

Goddard Space Flight Center
Greenbelt, Maryland

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Office of Technical Services, Department of Commerce,
Washington, D.C. 20230 -- Price $0.75

CODING AN ANALOG VARIABLE FOR CONSTANT PERCENTAGE ERROR

by
Rodger A. Cliff
Goddard Space Flight Center

SUMMARY

Systems are treated which code an analog variable as a sequence of discrete values. Given one of these discrete values, there is necessarily some uncertainty about what value of the analog input variable produced the discrete output since there are only a finite number of distinct outputs. This paper will investigate the nature of this uncertainty or possible error.

It is shown that a logarithmic quantization scheme produces an uncertainty in the output which is constant over the input range which the system is designed to cover. Relationships are derived between the uncertainty which must be tolerated, the width of the input range to be covered, and the number of discrete outputs required. These relationships are presented both analytically and graphically.

CONTENTS

CODING AN ANALOG VARIABLE FOR CONSTANT PERCENTAGE ERROR

by
Rodger A. Cliff
Goddard Space Flight Center

INTRODUCTION

An analog variable is one which may take on any value within a continuous range. In order to operate upon such a variable in a discrete device, such as a digital computer or a digital data transmission link, it is necessary to convert the analog variable to discrete form. In other words, the analog variable, which may have any of an infinite number of values, must be mapped onto a finite set of discrete values. This is shown schematically in Figure 1 and graphically in Figure 2. The device which accomplishes the mapping is referred to as an analog to digital converter. It repeatedly samples the analog input x and produces a discrete output y for each of these samples.

The analog variable is divided into intervals denoted A,B,C,... in Figure 2. An input x which falls within a given interval produces an output y corresponding to that interval. For instance, if the input x has a value within the interval B the analog to digital converter produces an output y having the discrete value b. Similarly, inputs within interval A produce an output a etc.

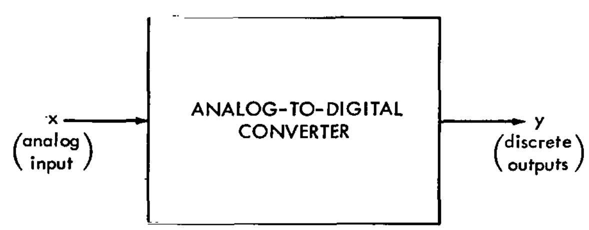

Figure 1 — Conversion of an analog variable to digital form.

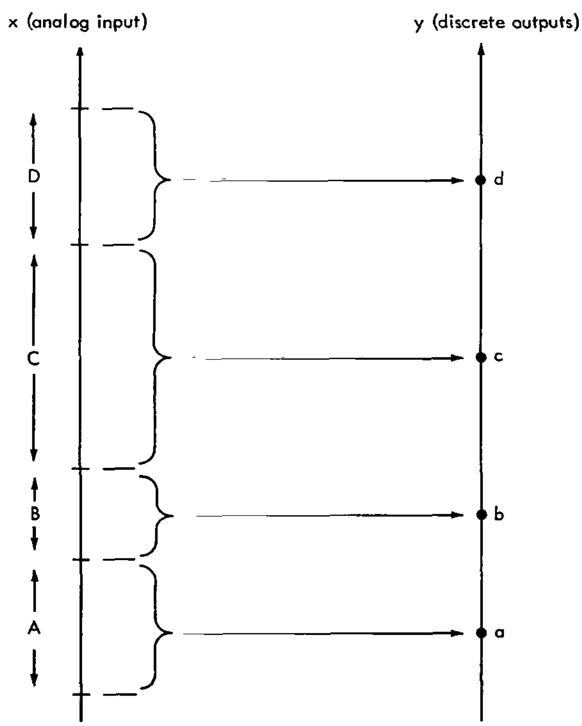

Figure 2 — Arbitrary quantization scheme.

There are various instances when it is necessary to convert a continuous variable into digital form: for example, analog input to a digital computer, or digital telemetry of continuous information. It is often desired that the possible percentage error associated with each discrete output be less than a certain amount. A logarithmic quantization is used to accomplish this end.

In this paper the problem of logarithmic quantization of a continuous input into a finite number of outputs is discussed. Results are presented showing the necessary number of discrete outputs for a given dynamic range and percentage error. For those interested in binary systems, a discussion of the number of bits required to express with a given accuracy an input variable of a given dynamic range is included.

LOGARITHMIC QUANTIZATION

Derivation of the Basic Relationships

A derivation of the relationship between the number of discrete outputs, the dynamic range, and the percentage error will be given. The relationship between the input x and the output y of a logarithmic quantization is depicted in Figure 3 on linear axes, and in Figure 4 on log axes. Let an arbitrary quantization interval X range from a lower value x^- to an upper value x^+. The output y_x corresponding to this interval is chosen to have a value between x^- and x^+. The choice is made such that the possible error is minimized.

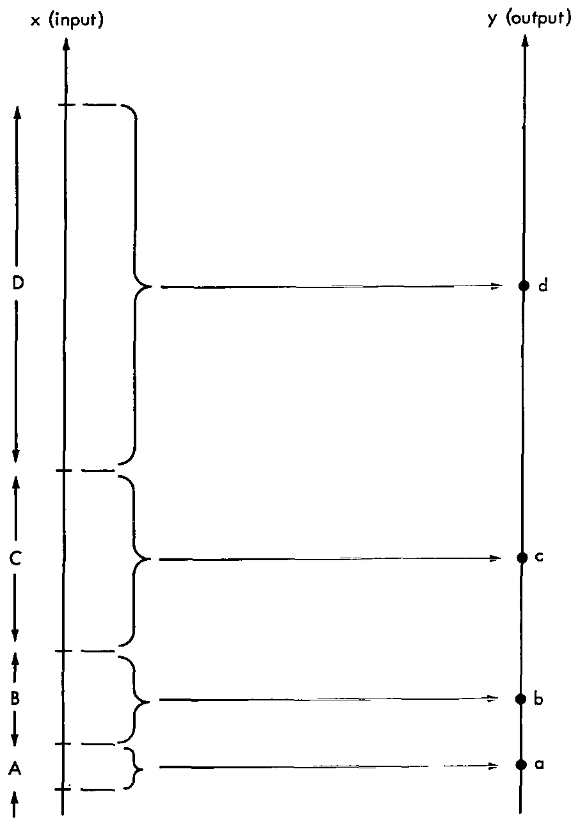

Figure 3 — Logarithmic quantization scheme — linear axes.

The maximum possible positive percentage error may be expressed as

$$Q^+ = 100 \frac{x^+ - y_x}{y_x}, \qquad (1a)$$

and the maximum possible negative percentage error as

$$Q^- = 100 \frac{y_x - x^-}{y_x}. \qquad (1b)$$

In order to obtain a symmetrical error bracket about y_x, we will choose y_x such that

$$Q^+ = Q^- = Q, \qquad (1c)$$

where Q is the maximum possible ± percentage error. The three Equations 1 may now be solved for y_x to obtain

$$y_x = \frac{x^+ + x^-}{2} . \tag{2}$$

It is not surprising that the optimum value for y_x is found to be halfway between x^- and x^+; certainly it is the conclusion we reach intuitively.

Equations 1 may also be solved for x^- and x^+.

$$x^+ = \left(1 + \frac{Q}{100}\right) y_x , \tag{3a}$$

$$x^- = \left(1 - \frac{Q}{100}\right) y_x . \tag{3b}$$

Observe that the ratio x^+/x^- of the upper limit to the lower limit of the quantization interval X is independent of which interval X may be. This ratio shall be denoted by K:

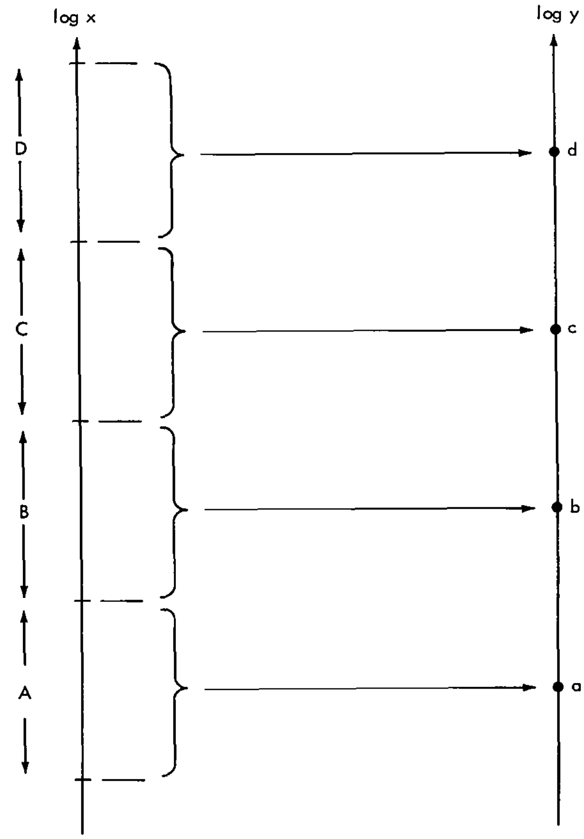

Figure 4 — Logarithmic quantization scheme — log axes.

$$K = \frac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}} = \frac{x^+}{x^-} . \tag{4}$$

Equation 4 says that for any interval X, $x^+ = K x^-$; therefore the levels that separate intervals are expressible in terms of the lower limit of the lowest interval, as shown in Figure 5.

The dynamic range R is defined to be the ratio of the largest possible value of the input variable x to the smallest possible value of the input variable x that fall within the range to be quantized. From Figure 5 we see that

$$R = \frac{K^N x_0}{x_0} = K^N , \tag{5}$$

where N is the number of quantization intervals.

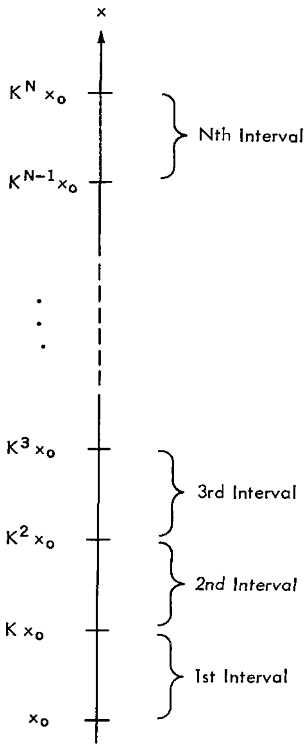

Figure 5 – Decision levels for a logarithmic quantization.

To obtain N as a function of K (and therefore as a function of Q), we take the logarithms of both sides of Equation 5:

$$\log R = N \log K , \tag{6}$$

or

$$N = \frac{\log R}{\log K} , \tag{7}$$

which in terms of Q is

$$N = \frac{\log R}{\log \left(\dfrac{1 + \dfrac{Q}{100}}{1 - \dfrac{Q}{100}} \right)} . \tag{8}$$

This is the result we require, relating N (the number of quantization intervals required) to R (the dynamic range) and Q (the maximum possible ± percentage error).

The Number of Bits Needed for a Binary System

A binary code of B bits can specify that one out of 2^B possible events has occurred. In the system under discussion, a particular event occurs when the input variable falls within a particular one of the N quantization intervals. Therefore, there must be at least as many characters in the binary code as there are quantization intervals. That is

$$2^B \geq N , \tag{9}$$

so that

$$B \geq \log_2 N . \tag{10}$$

It is advantageous that N be a power of 2; then a binary code may be chosen such that each character is used and full advantage is taken of the capacity of the binary system. In this case there are N quantization intervals and N characters in the binary code. The relationship (Equation 10) will hold as an equality.

APPLICATION OF THE RESULTS OF THE DERIVATION

Graphs of the Basic Equation

To facilitate system design, the basic relationship of the logarithmic quantization scheme (Equation 8) has been plotted. Figure 6 shows the trade-off between the dynamic range and the number of intervals required for a given percentage error. Since the number of intervals N varies as the logarithm of the dynamic range R for fixed percentage error Q, the equation plots as a family of straight lines on semi-log coordinates.

Figure 7, on the other hand, emphasizes the relationship between the number of intervals needed and the percentage error. It may be shown that N varies approximately as 1/Q (see Appendix A); hence contours of constant R are straight lines on full logarithmic coordinates.

General Remarks

A few general remarks are in order at this point. First, notice that for the condition where the percentage error is greater than about 5 percent, the number of quantization intervals required for a given dynamic range is relatively insensitive to the percentage error (see Figure 6). However, as the percentage error approaches zero the number of intervals required increases rapidly. Notice also that the number of intervals needed is much more sensitive to changes in the dynamic range when the dynamic range is small than it is when the dynamic range is large.

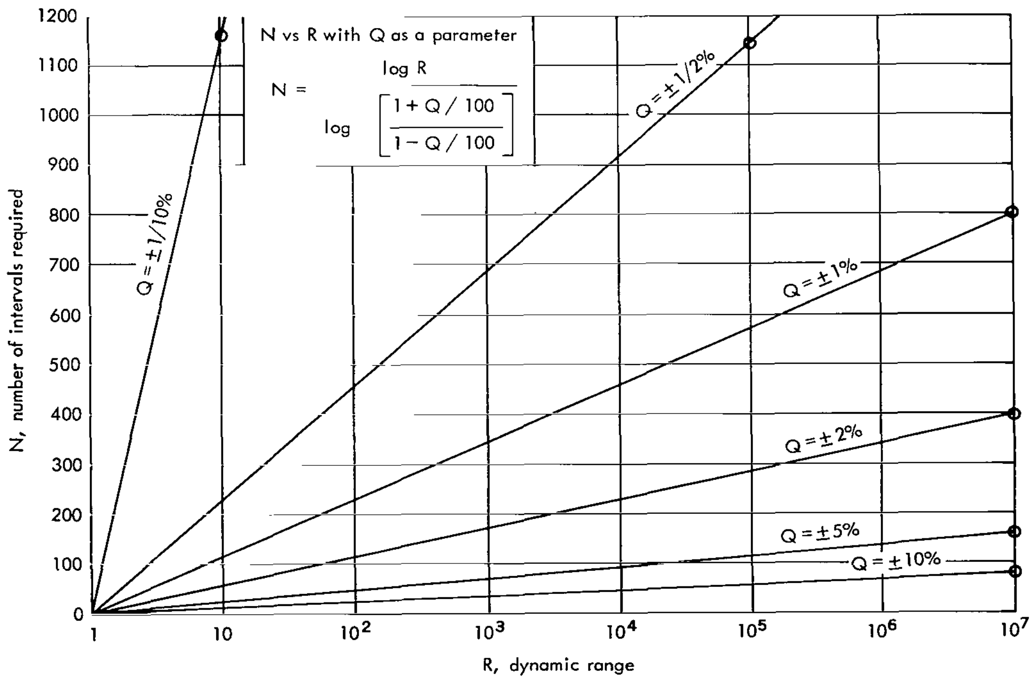

Figure 6 — The trade-off between the dynamic range and number of intervals required.

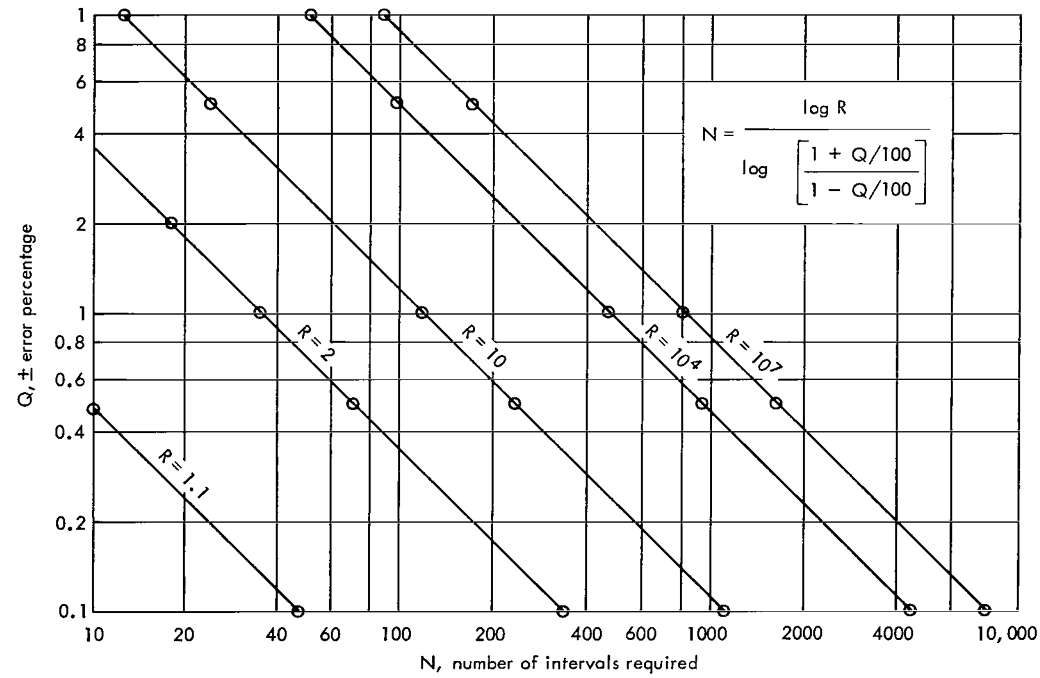

Figure 7 — Relationship between number of intervals required and percentage error.

~~Figure 7 shows that a change of dynamic range from 2 to 10 (a factor of 5) increases the number~~ of intervals needed in about the same proportion as a change in dynamic range of from 10 to 10,000 (a factor of 1,000).

Application to Binary Systems

Binary system parameters are plotted in Figures 8, 9, and 10. The basic relationship governing a binary system is obtained by combining Equation 8 with the inequality (Equation 9) to obtain

$$2^B \geqq \frac{\log R}{\log \left(\frac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}}\right)} . \tag{11}$$

The number of bits required is the smallest integer B such that Equation 11 is satisfied.

Turning our attention to Figure 8, we see the number of bits B versus the error Q for various values of the dynamic range R. This is a particularly useful presentation of system parameters, because the dynamic range is a characteristic of the input variable, and as such is often not under the control of the designer of the coding system.

Since B must be an integer, the curves are discontinuous and have a stair-step appearance. The most desirable points on these curves are the circled lower corner points; at these points

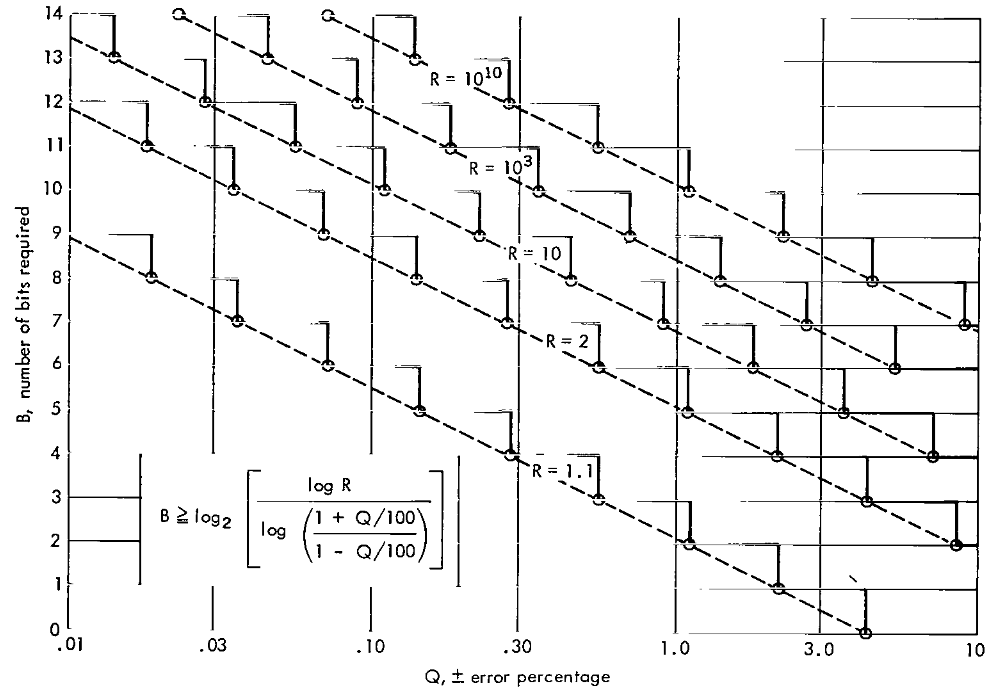

Figure 8 — The binary system parameters B and Q.

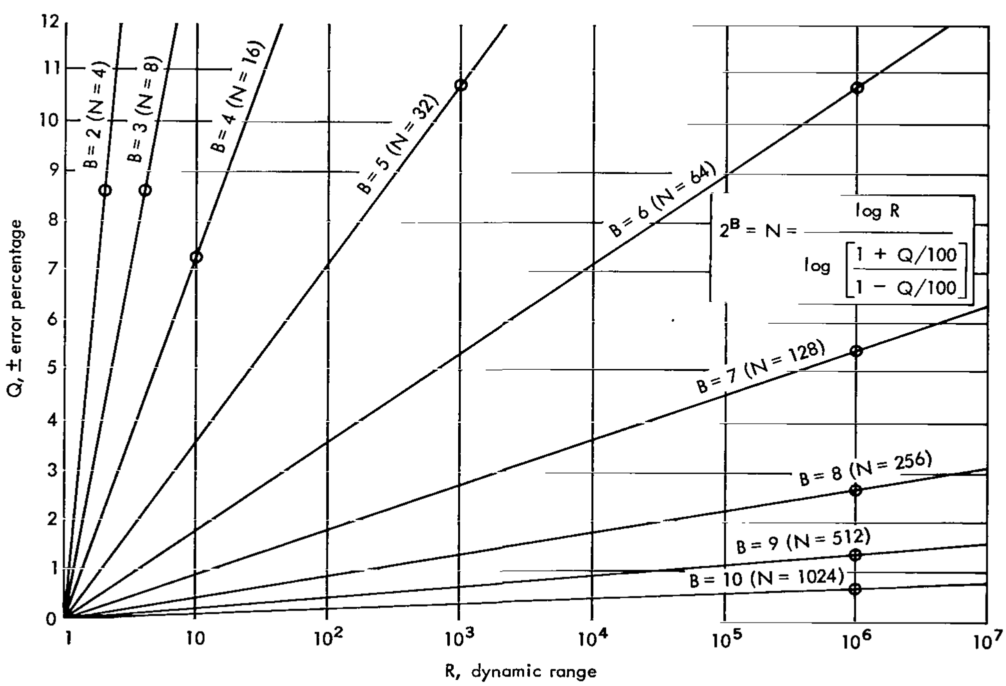

Figure 9 — The binary system parameters Q and R.

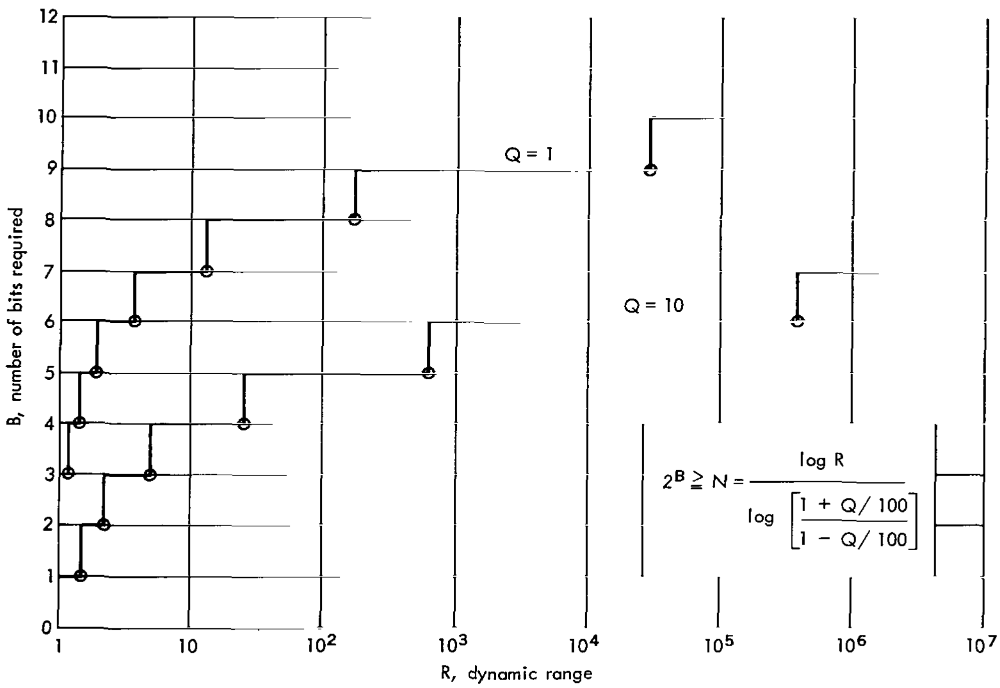

Figure 10 — The binary parameters B and R.

Equation 11 holds as an equality and full advantage is being taken of the capacity of the binary system. At these points one obtains the minimum percentage error for any given dynamic range and number of bits.

If, on the other hand, the designer is confronted with a system of fixed binary capacity, and he wishes to know the percentage error versus the dynamic range, Figure 9 will be found useful. In this figure the percentage error Q is plotted versus the dynamic range R for binary systems of from 2 to 10 bits.

For fixed percentage error, the trade-off between the number of bits and the dynamic range is shown in Figure 10. Again, the desirable operating points are the circled lower corner points where full use is made of the capacity of the binary system. It is helpful to use this graph when trying to decide just how wide a dynamic range to cover when the exact range of the input variable is not known and some guard space is to be left at the ends of the dynamic range.

METHODS OF ACHIEVING LOGARITHMIC QUANTIZATION

There are a number of methods which may be used to effect logarithmic quantization. Conceptually, the most appealing method is to use an analog to digital converter which has logarithmically spaced decision levels. This is the sort of device described heretofore. An equivalent result is produced if the logarithm of the input x is taken using analog techniques, and then the logarithm is subsequently quantized by a linear analog-to-digital converter. It is also possible to convert an analog input to digital form and to then take the logarithm using digital methods.*

*D. H. Schaefer, Goddard Space Flight Center, has developed simple methods of computing logarithms digitally.

Unfortunately this approach requires a rather large capacity digital system preceeding the point where logarithms are computed. Particularly, there may be problems involved in obtaining a linear analog-to-digital converter which has both the required dynamic range and the required accuracy in the lower portion of the dynamic range.

AN EXAMPLE

A Logarithmic System

It will be instructive to consider an example of a system using logarithmic quantization and to compare this to linear systems that might be used in the same application. We will assume that the input variable has values of interest which range from 1 to 10, and that we wish to obtain ± 1 percent accuracy. In the example then, R = 10 and Q = 1. The number of bits that will be required may be determined from Figure 8 by following the Q = 1 line up from the Q axis until it intersects the plot for R = 10. The intersection is found to lie on the B = 7 line; hence 7 bits will be required. Observe that the system will not be optimum in the sense that with 7 bits (128 quantization intervals) either less error or more dynamic range could be obtained. Reference to Figure 8 reveals that in a 7 bit system with a dynamic range of 10, the error need be only ±.9 percent. On the other hand, the Q = 1 contour in Figure 10 shows that for 7 bits the dynamic range may be as great as 13 without exceeding ±1 percent error.

In our example, we will cover the 10 to 1 dynamic range with sufficient intervals to give ±1 percent error. This will require

$$N = \frac{\log R}{\log\left(\dfrac{1+\frac{Q}{100}}{1-\frac{Q}{100}}\right)} = \frac{\log 10}{\log\left(\dfrac{1.01}{.99}\right)} = 116 \text{ intervals.}$$

Since $2^6 = 64$ and $2^7 = 128$, we require 7 bits. Since 128 - 116 = 12, 12 characters of the 7 bit binary code will not be required. These characters can be used to indicate various error conditions, such as an input which was not within the prescribed dynamic range or a malfunction of the electronic circuitry.

Two Linear Systems

If we do not wish to employ a logarithmic system, an obvious alternative is a linear system. Two such systems come immediately to mind. One has a possible error of ±1 percent at full scale and progressively greater error as the value of the input variable decreases. The other will have ±1 percent error at the lower end of the dynamic range and considerably less at the upper end. These represent extremes between which any system which is roughly equivalent to our logarithmic system will fall.

Consider first the case where the error is to be ±1 percent at full scale. As before, the dynamic range will be 10. Starting at full scale (which we will take to be 10) the first few quantization intervals and the outputs which correspond to them are shown at the top of Figure 11. At the other end of the dynamic range, the last few quantization intervals are also shown. For the interval $9.8 < x < 10.0$, the percentage error is $Q'_{fs} = \pm(.1/9.9)\,100 = \pm 1.01$ percent; at the low end of the scale $1.0 < x < 1.2$, $Q'_{bs} = \pm(.1/1.1)\,100 = 9.1$ percent. Observe that the width of any quantization interval is 0.2; therefore $(10-1)/.2 = 45$ intervals will be required for this system.

Since $2^5 = 32$ and $2^6 = 64$, 6 bits will be required, thus $64 - 45 = 19$, and 19 characters of the binary code are not used. As was the case with the logarithmic system, these characters may be used to extend the dynamic range or reduce error, or they may serve as indication of malfunctions and error conditions.

The second linear system is shown in Figure 12. Again it will be assumed that the input ranges from 1 to 10. For the interval $1.00 < x < 1.02$, $Q''_{bs} = (.01/1.01)\,100 = .99$ percent. As the

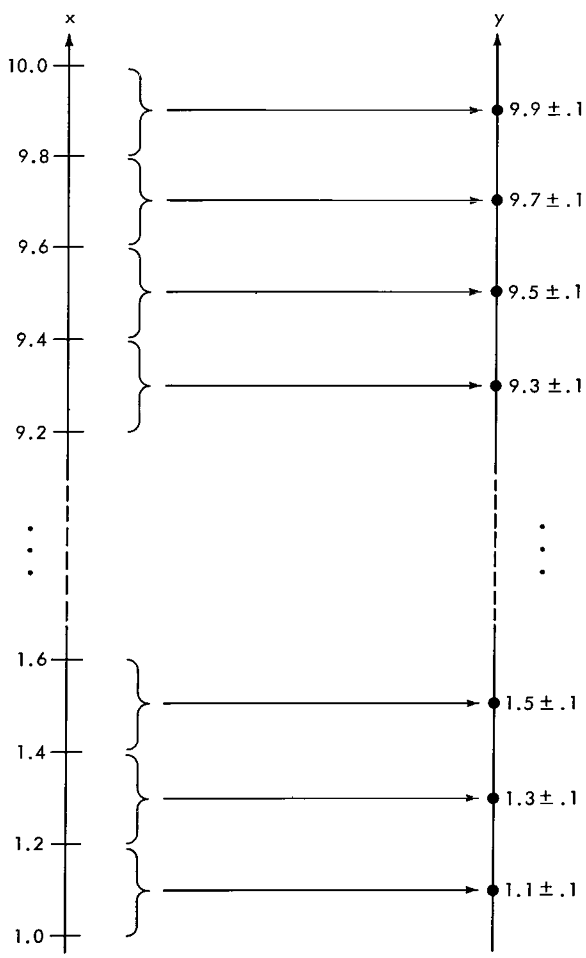

Figure 11 — Linear system — Error ± 1 percent of full scale.

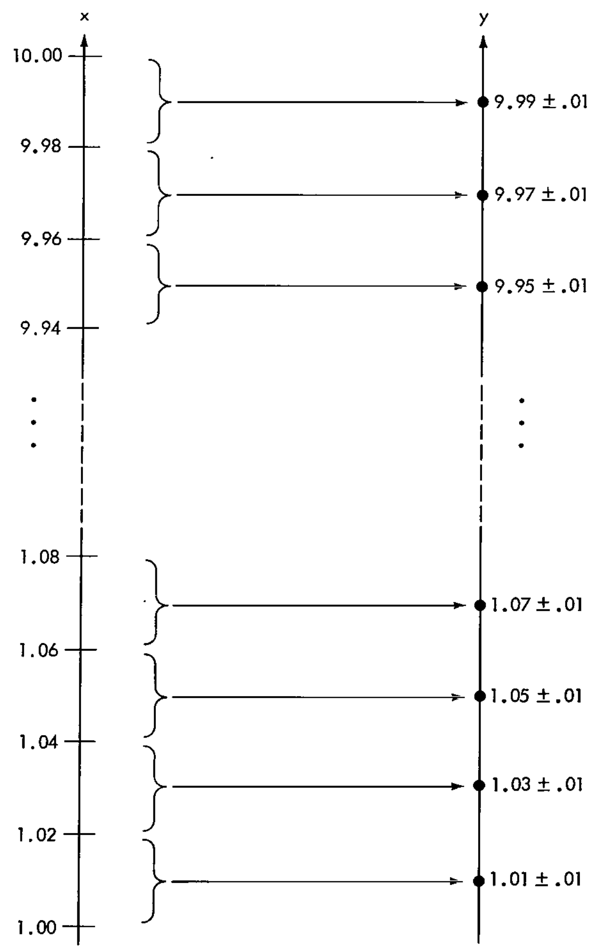

Figure 12 — Linear system — Error ± 1 percent of bottom scale.

value of the input variable increases, the error decreases until for the interval $9.98 < x < 10.00$, $Q''_{fs} = (.01/9.99)\,100 = .10$ percent. Each interval has the width .02; so $(10 - 1)/.02 = 450$ intervals will be required. $2^8 = 256$ and $2^9 = 512$; therefore 9 bits must be used in the binary system. Thus $512 - 450 = 62$ unused characters. As before, they may be used to extend the dynamic range, reduce the error, or indicate malfunctions and error conditions.

Comparison of Logarithmic and Linear Systems

The quantization intervals of the logarithmic system are shown in Figure 13. Notice that near $x = 10$ they resemble those of the first linear system, but near $x = 1$ they resemble those of the second linear system.

Table 1 gives a comparison of the logarithmic system and the two linear systems. In order to achieve a percentage error which is everywhere at least as small as that of the logarithmic system, a linear system (System C) requires over three times as many quantization intervals. However, System C provides more accuracy than is needed throughout most

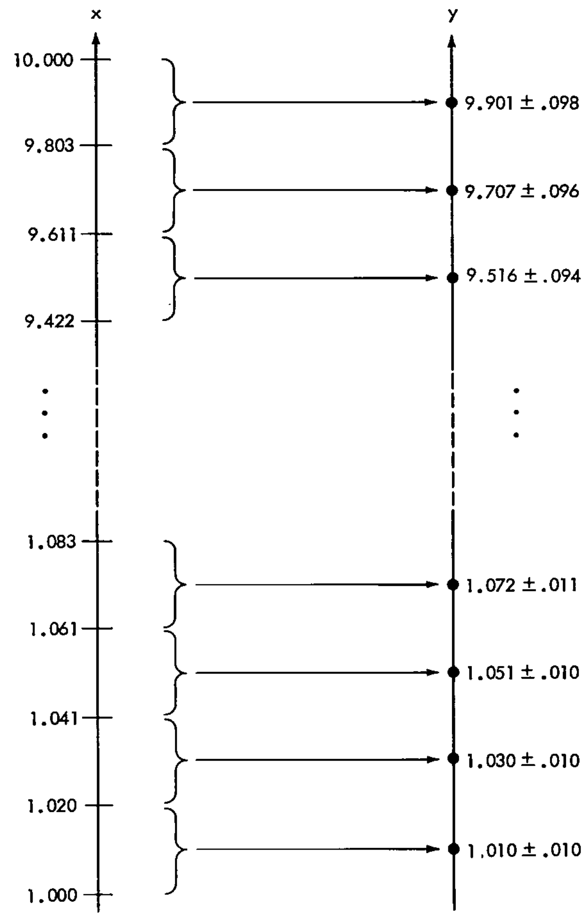

Figure 13 — Logarithmic system — Error ± 1 percent of output.

Table 1

The Characteristics of Three Systems.

Characteristic	System A: Linear, ±1% of greatest output	System B: Log, ±1% of output	System C: Linear, ±1% of least output
Greatest Input	10	10	10
Least Input	1	1	1
Dynamic Range	10	10	10
Greatest Output	9.9 ± .1	9.9 ± .1	9.99 ± .01
Least Output	1.1 ± .1	1.01 ± .01	1.01 ± .01
Width of Quantization Intervals	.2	.02y	.02
Number of Quantization Intervals	45	116	450
Number of Bits Required	6	7	9
Unused Characters in Binary Code	19	12	62
Total Characters in Binary Code	64	128	512

of the dynamic range. This linear system is inefficient because it provides more information than is desired. System A, on the other hand, uses about 1/3 as many quantization intervals as the logarithmic system. Unfortunately it does not meet the error specifications except at full scale. It is undersirable because it does not provide enough information.

Although in the example the dynamic range of 10 ran from 1.0 to 10.0, the percentage error and the number of intervals required are a function only of the dynamic range, and not of the absolute value of the limits of the dynamic range. This applies to all three systems which were considered.

It should be remarked that there are situations which require high accuracy in only a portion of the dynamic range. The remaining part of the dynamic range can be covered by relatively wide quantization intervals. These systems must be tailored to specific applications.

There are other instances where the error is specified to be ± so many units, rather than as a percentage. In this case a linear system would be used. For example, system A provides an error of ± .1. The logarithmic system also meets this criterion, but it uses three times as many quantization intervals. Error would be specified in this way if for example the difference of two discrete outputs of near equal magnitude was to be calculated.

CONCLUDING REMARKS

It has been shown that a logarithmic system can be used to advantage to code an analog variable as a sequence of discrete outputs. This coding is such that the possible error associated with each discrete output is the same. An example has shown that if a system is specified as having less than a certain percentage error throughout its dynamic range, that a logarithmic system is considerably more efficient than a linear system.

The basic equations describing the logarithmic system are repeated here for convenience:

$$N = \frac{\log R}{\log\left(\dfrac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}}\right)}, \tag{8}$$

$$\frac{x^+}{x^-} = \frac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}}, \tag{4}$$

$$y_x = \frac{x^+ + x^-}{2}. \tag{2}$$

Appendix B contains the definitions of the symbols used in these equations and elsewhere in this paper.

A logarithmic system may involve more complex equipment for the conversion of an analog variable to discrete form than a linear system would. However, the digital system which uses the discrete outputs of the logarithmic system will have fewer bits upon which to operate (for equal percentage errors) and will consequently be less complex. This feature is particularly attractive in the telemetry business where it is frequently necessary to transmit a maximum of information with a minimum number of bits in order to conserve bandwidth and transmitter power.

(Manuscript received October 3, 1963)

Appendix A

Derivation of the 1/Q Relationship

It may be shown that for constant R, N varies as 1/Q. In the case of constant R, Equation 9 becomes

$$N = \frac{\log_a R}{\log_a \left(\dfrac{1 + \dfrac{Q}{100}}{1 - \dfrac{Q}{100}} \right)} = \frac{\text{const}}{\log_a \left(\dfrac{1 + \dfrac{Q}{100}}{1 - \dfrac{Q}{100}} \right)} . \tag{A1}$$

The logarithms in the right-hand terms may be taken to any convenient base. By the properties of logarithims,

$$\log \left(\frac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}} \right) = \log \left(1 + \frac{Q}{100}\right) - \log \left(1 - \frac{Q}{100}\right) . \tag{A2}$$

The left hand terms of Equation A2 may conveniently be expanded in a power series by application of the relation:*

$$\log_e (1 + X) = X - \frac{1}{2} X^2 + \frac{1}{3} X^3 - \frac{1}{4} X^4 + \cdots . \tag{A3}$$

This yields

$$\log_e \left(1 + \frac{Q}{100}\right) = \frac{Q}{100} - \frac{1}{2} \left(\frac{Q}{100}\right)^2 + \frac{1}{3} \left(\frac{Q}{100}\right)^3 - \frac{1}{4} \left(\frac{Q}{100}\right)^4 + \cdots , \tag{A4}$$

$$- \log_e \left(1 - \frac{Q}{100}\right) = \frac{Q}{100} + \frac{1}{2} \left(\frac{Q}{100}\right)^2 + \frac{1}{3} \left(\frac{Q}{100}\right)^3 + \frac{1}{4} \left(\frac{Q}{100}\right)^4 + \cdots . \tag{A5}$$

Hence,

$$\log_e \left(\frac{1 + \frac{Q}{100}}{1 - \frac{Q}{100}} \right) = 2 \left(\frac{Q}{100}\right) + \frac{2}{3} \left(\frac{Q}{100}\right)^3 + \cdots ; \tag{A6}$$

Handbook of Chemistry and Physics, 43rd ed., pg. 323.

and therefore, if $\frac{Q}{100} << 1$,

$$N \approx \frac{\log_e R}{2 \frac{Q}{100}} = \frac{\text{const}}{Q} . \tag{A7}$$

By using the relation*

$$\log_e x = 2.30 \log_{10} x , \tag{A8}$$

Equation A7 may also be written

$$N \approx \frac{115 \log_{10} R}{Q} . \tag{A9}$$

Handbook of Chemistry and Physics, 43rd ed., pg. 12.

Appendix B

Symbols

B	number of bits required for N intervals
$K = x^+/x^-$,	the dynamic range covered by one quantization interval
N	number of quantization intervals required
Q^+	maximum possible positive percentage error
Q^-	maximum possible negative percentage error
Q	maximum possible ± percentage error
Q'_{bs}	bottom of scale percentage error for system A
Q'_{fs}	full scale percentage error for system A
Q''_{bs}	bottom of scale percentage error for system C
Q''_{fs}	full scale percentage error for system C
R	dynamic range over which the input variable is to be quantized
x	analog input variable
X	an arbitrary quantization interval
x	lower limit of quantization interval X
x^+	upper limit of quantization interval X
y	discrete output variable or system C
y_x	value of output variable corresponding to quantization interval X